AMENDMENTS TO THE SPECIFICATION

Please replace the paragraph beginning at page 3, line 12, with the following amended paragraph:

These and other needs are addressed by the various embodiments and configurations of the present invention. The present invention generally matches or associates session packets communicated in a session between two or more endpoints or participants with the identities of the participants, *e.g.*, session ids (e.g., SSRC) and/or network addresses (e.g., transport addresses), creating new sessions if appropriate. Each session participant is typically identified by network address (*e.g.*, *UDP* port and Internet Protocol address) and/or session *(e.g.*, SSRC) [[id]].

Please replace the paragraph beginning at page 5, line 18, with the following amended paragraph:

The various embodiments of the present invention can have numerous advantages. For example, the window of opportunity for confusing concurrent sessions and attributing data in RTCP packets to the wrong session can be much smaller than with current architectures. The window of opportunity for possible confusion using the above algorithm(s) exists only when two different endpoints join different sessions at the same time and with the same SSRCs [[ids]]. This window of opportunity or startup interval closes once either of the endpoints (or their peers) has sent an RTCP packet with a reception block corresponding to either endpoint. Once the reception block is exchanged, the SSRCs [[ids]] of both parties to the session are known to the monitor. Before such an exchange, the monitor typically has only the SSRC [[id]] and network address of one party to the session. The SSRC [[id]] and network address of the other party is unknown. The startup interval is typically fairly short, *e.g.*, typically on the order of 5 seconds or less. Even this potential period of confusion can be eliminated by choosing an SSRC [[id]] for each endpoint that is globally unique. The use of the active session table and network address to define the session (rather than only pairings of SSRCs [[ids]]) can, after the startup interval, at

least substantially eliminate misinterpretation of RTCP packets and incorrect analysis of performance data. The accuracy of the algorithm(s) in matching RTCP packets with the corresponding session results in more accurate statistical analysis of the communication link in the network.

Please replace the paragraph beginning at page 8, line 14, with the following amended paragraph:

The operations of the matching algorithm(s) in the monitor 300 will now be discussed with reference to Fig. 2. Referring to Fig. 2, a packet is received by the monitor in step 200. Parser 312 parses the packet to locate selected fields, which typically are the source transport address, source SSRC [[id]] ("the endpoint SSRC [[id]]"), if present the destination transport address of the other session participant (which is possibly in the application APP field), and, if present, the destination SSRC [[id]] of the other session participant in the receiver report blocks (the SSRC's in the receiver report blocks are hereinafter referred to as the "reception report SSRC [[id]]"). As will be appreciated, the reception report is typically a report regarding the characteristics of the communication link, such as the condition of the voice stream experienced since the last reception report.

Please replace the paragraph beginning at page 10, line 7, with the following amended paragraph:

If the matcher 316 receives a hit, the monitor in step 224 updates the entry for the orphan session. This is typically done by updating the other party's SSRC [[id]] (if available) and updating the associated data in the packet. As noted, each entry in the orphan session table includes at least UDP or transport address of an endpoint, an endpoint SSRC [[id]], and optionally reception report SSRC [[id]].

Please replace the paragraph beginning at page 10, line 20, with the following amended paragraph:

Fig. 4 depicts the algorithm for a computational component in the first endpoint that is configured to input another (second) endpoint's SSRC [[id]] into the APP field. In step 400, the first endpoint receives an RTCP packet from the second endpoint participating in a session with the first endpoint. In step 404, the first endpoint parses the RTCP packet and determines whether a flag has been set (*i.e.*, determines the flag's value). The flag identifies whether or not the second endpoint is configured to transmit a duplicate packet to the session monitor. If the flag is set (meaning that the second endpoint is configured to transmit a duplicate packet to the session monitor), the first endpoint in step 408 does not forward a modified version of the RTCP packet to the monitor. The first endpoint returns to step 400 to await the next RTCP packet. If the flag is not set (meaning that the second endpoint is not configured to send a duplicate RTCP packet to the monitor), the first endpoint in step 412 modifies the RTCP packet by replacing the destination address with that of the monitor and inputs into the APP field the second endpoint's network address and forwards the modified packed to the monitor 300. The forwarding can be done by any suitable technique such as port forwarding.